# 1. Madison

### Program Name: Madison.java Input File: madison.dat

Madison always tells the truth, well, sometimes. She uses the words "and", "or", "not", and "xor" when talking about two things at the same time. For example she would say, "I took out the trash and fed the dog". If indeed she did both things, then she would be telling the truth. However, if she really did not feed the dog, then she would not be telling the truth.

Now if she said, "I took out the trash or I fed the dog.", she would be telling the truth overall if she did indeed feed the dog, but did not take out the trash. As long as at least one of those two situations was true, she would be telling the truth.

What's really confusing is when she says, "xor". This means that she is telling the truth when only one of the two things is true. For example, if she says, "I fed the dog xor I cleaned my room.", she would only be telling the truth if she only did one of those two things. If she did both, it would be a lie, or if she did neither, that would also be false.

We'll represent two or three situations using the letters A, B and C, in some kind of combination expression using NOT(!), AND(*), XOR(^) and OR(+), in that order of priority. In other words, NOT has top priority, and OR has lowest priority. You'll also know the true or false state of each letter.

For example, for the expression A+B, and the values 11, which means A is 1 and B is 1, the final result will be true, because true OR true is true.

For the expression A*B, and the values 10, the final result would be false, since true AND false is false.

There may be parentheses involved as well, in which case the expression inside would be evaluated first. There will be no instances of nested parentheses. It is possible for any of the variables to be used more than once in an expression, like A*B+A*C.

**Input:** Several lines of data, each line with an expression containing no spaces, followed by a two or three digit string of zeros and ones, representing the corresponding values of the variables in the expression.

**Output:** The final true or false value of the expression, given the actual values of each variable.

**Sample Input:**

```
A+B 11
A*B 10
A+B*C 101
A^B 01
!(A+B) 10
```

**Sample Output:**

```
true
false
true
true
false
```